

DETAILED ACTION

1. Claims 1-66 are presented for examination.

Specification

2. The disclosure is objected to because of the following informalities: at page 5, lines 3-4, the code is referred to as logic 204, however, at lines 9-10, "logic 203" is used, which should be "logic 204".

Appropriate correction is required.

Claim Objections

3. Claims 4, 5, 29, 37, 38 and 62 are objected to because of the following informalities:

Claim 4 recites "the set comprising: adding; clearing, creating a view, inserting, removing, reversing; setting a range; and sorting" in lines 2-3. Examiner notes that ";" and "," are used when listing elements of the set and does not know the distinguish between the two. More explanation is requested for further consideration.

Claim 5 recites "source data" on line 2, "a" should be used before "source data".

Claim 27 recites "handing a CurrentChanged event", "handling" should be used instead of "handing".

Claim 29 recites "one of the following: items within a collection have changed: an item has been added, removed, or the entire collection has been refreshed" in lines 2-3, which uses ":" twice which makes it unclear as which one belong to the collection.

Claims 37, 38, 60 and 62 suffer the same problems as claims 4, 5, 27 and 29 above, respectively.

Appropriate corrections are required.

Claim Rejections - 35 USC § 112

4. The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

5. Claims 1-66 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

Claim 1 recites "A method for providing an interface for a function", however, the rest of the claim does not teach how an interface is provided for a function.

Claim 4 recites the limitation "the set" in line 2. There is insufficient antecedent basis for this limitation in the claim.

Claim 6 recites the limitation "the bind declaration object" in lines 1-2. There is insufficient antecedent basis for this limitation in the claim.

Claim 15 recites "serving as a data source for data binding" in lines 1-2, which is unclear as what being served as data source, because performing an operation is not a data source.

Claim 20 recites "receiving a list of sql commands and names of tables that they should be used to fill" which is unclear as what information the limitation tried to convey, furthermore, "they" refers to commands or tables, and "fill" what, tables or something else?

Claims 2-33 and 35-66 fail to remedy deficiencies of claims 1 and 33 above.

Claim Rejections - 35 USC § 101

6. 35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

Claims 34-66 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter. The claims are directed to a signal directly or indirectly by claiming a medium and the Specification recites evidence where the computer readable medium is define as a “*wave*” (such as a carrier wave), see Specification, page 4, lines 16-27. In that event, the claims are directed to a form of energy which at present the office feels does not fall into a category of invention.

Claim Rejections - 35 USC § 102

7. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

8. Claims 1-16, 18, 24, 26-29, 31, 33-49, 51, 57, 59-62, 64 and 66 are rejected under 35 U.S.C. 102(e) as being anticipated by Hayton et al. (U.S. 7,194,743 B2).

As to claim 1, Hayton teaches a method for providing an interface for a function, comprising:

- receiving at least one parameter (If the user updates a UI element, the UI element 46 signals a change event to the event manager; col. 19, lines 12-14), and
- in response to receiving the at least one parameter (When the event manager 74 receives a change event from the UI element 46; col. 19, lines 14-16), performing an operation relating to a binding on at least one property from at least one of data sources, data source classes and data-specific implementations of collections and views (the event manager determines, using the mapping, ... server portion 22b can update the associated property 38 of the application component 34 in response to the user change of the UI element 46; col. 19, lines 16-25).

As to claim 2, Hayton teaches wherein performing the operation, further comprises performing handling collection changed events in data collection underlying a collection view (Property paths are not wired ... to reflect the current state of property 38e; col. 12, lines 40-59).

As to claim 3, Hayton teaches wherein performing the operation, further comprises performing a function relating to array list data collection (The 'HRApp' portion of the property path ... whoever that employee may now be; col. 13, line 48 - col. 14, line 12).

As to claim 4, Hayton teaches wherein performing the function relating to the array list data collection, further comprises performing at least one of the functions within the set

comprising: adding, clearing, creating a view, inserting, removing, reversing, setting a range, and sorting (a property path can also represent ... pointing to the salary property of the twenty-seventh employee, regardless of who that may be; col. 13, lines 13-29).

As to claim 5, Hayton teaches wherein performing the operation, further comprises managing bindings between a dynamic property user interface and a source data (dynamically binding property path ... when instances of application components 34 are created or deleted; col. 17, lines 1-6 and The client-side event manager ... the server portion 22b also communicates changes to the client portion 22a in terms of complete property path; col. 17, lines 40-65).

As to claim 6, Hayton teaches wherein performing the operation, further comprises getting the bind declaration object of a binding expression (inherent from "the event manager 74 determines, using the mapping"; col. 19, lines 16-17 and col. 17, lines 40-65 and col. 11, lines 49-64).

As to claim 7, Hayton teaches wherein performing the operation, further comprises performing a collection view function (The event manager 74 communicates the updates due to the change event to each of the other UI elements 46 mapped to the same property path; col. 19, lines 18-20).

As to claim 8, Hayton teaches wherein performing the operation, further comprises implementing a collection view that includes checks for context infinity (In other embodiment,

the event manager does not make ...that the user changed back to its pre-user intervention state; col. 19, lines 26-44).

As to claim 9, Hayton teaches wherein performing the operation, further comprises supporting object references to objects being used as data context for a binding (object pointer; col. 10, lines 55-65).

As to claim 10, Hayton teaches wherein performing the operation, further comprises encapsulating arguments for data transfer events, wherein the events are routed event that are handled by a designated handler based on a delegate (The system 54 uses a data compression scheme ... to use a value 50 from its current value table; col. 28, lines 27-43).

As to claim 11, Hayton teaches wherein performing the operation, further comprises handling a data transfer event raised by a binding (The event manager 74 communicates the updates due to the change event to each of the other UI elements 46 mapped to the same property path; col. 19, lines 18-20).

As to claim 12, Hayton teaches wherein performing the operation, further comprises representing an object reference to an element, with the object reference being specified by its element ID (S->C App.Employee[0].Name=Alf; col. 28, lines 10-20).

As to claim 13, Hayton teaches wherein the object reference is an explicit object reference (object pointer; col. 10, lines 55-65).

As to claim 14, Hayton teaches wherein performing the operation, further comprises implementing a collection view for collections based on a list (See Fig. 5 and the page 42" contains a tree view type UI element; col. 22, lines 1-47).

As to claim 15, Hayton teaches wherein performing the operation, further comprises serving as a data source for data binding (application components; col. 21, lines 51-64).

As to claim 16, Hayton teaches wherein performing the operation, further comprises holding a collection of named parameters (col. 22, lines 37-47).

As to claim 18, Hayton teaches wherein performing the operation, further comprises encapsulating arguments passed in an event relating to at least one of an ObjectDataSource, and RefreshCompleted event of an XMLDataSource (The server portion 22b ... must be notified; col. 18, lines 57-65).

As to claim 24, Hayton teaches wherein performing the operation, further comprises serving as a data source for data binding to Extensible Markup Language (XML) content nodes (Application components 34 can be representations of nodes within a structured data file, such as XML; col. 10, lines 20-26).

As to claim 26, Hayton teaches wherein performing the operation, further comprises managing a view of a data collection (see Fig. 5 and col. 22, lines 1-36).

As to claim 27, Hayton teaches wherein performing the operation, further comprises handling a CurrentChanged event raised by collection views (When the event manager 74 receives a change event from the UI element 46; col. 19, lines 14-16, the event manager determines, using the mapping, ... server portion 22b can update the associated property 38 of the application component 34 in response to the user change of the UI element 46; col. 19, lines 16-25), or any class implementing the ICurrentItem interface. It is noted that meets only one of the two will meet the claim.

As to claim 28, Hayton teaches wherein performing the operation, further comprises representing a method that handles a CurrentChanging event raised by collection view classes, or any class implementing the ICurrentItem interface (The server portion 22b ... must be notified; col. 18, lines 57-65).

As to claim 29, Hayton teaches wherein performing the operation, further comprises enabling notification from at least one of the following: items within a collection have changed: an item has been added, removed, or the entire collection has been refreshed (A page 42 can also be altered ... more complex set of UI elements; col. 19, line 57 - col. 20, line 11).

As to claim 31, Hayton teaches wherein performing the operation, further comprises, maintaining a concept of the current record pointer in a collection view (the client portion 22a ... property path manager; col. 23, lines 6-23).

As to claim 33, Hayton teaches wherein performing the operation, further comprises describing a sort qualifier that is used to sort items in a collection when creating a view (If the table is sorted; col. 14, lines 28-32).

As to claim 34, it is the same as the method claim of claim 1 except this is a computer product claim and is rejected under the same ground of rejection.

As to claims 35-49, 51, 57, 59-62, 64 and 66, see rejections of claims 2-16, 18, 24, 26-29, 31 and 33 above, respectively.

Claim Rejections - 35 USC § 103

9. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

10. Claims 19, 21, 32, 52, 54 and 65 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hayton et al. (U.S. 7,194,743 B2).

As to claim 19, Hayton does not explicitly teach wherein performing the operation, further comprises handling events relating to at least one of a `ObjectDataSource.RefreshCompleted` event and a `XmlDataSource.RefreshCompleted` event. However, Hayton teaches handling events relating to change in object data source (server portion 22b ... must be notified; col. 18, lines 57-65). It would have been obvious to one of ordinary skill in the art that the system of Hayton can be implemented to have event specified as `ObjectDataSource.ReferhCompleted` event because it is just a different way to implement in the art.

As to claim 21, Hayton teaches getting data from a database for use in databinding (Application components 34 can be representations of nodes within a structured data file, such as XML or a database; col. 10, lines 20-26). However, Hayton does not explicitly teach a SQL server. It would have been obvious to one of ordinary skill in the art at the time the invention was made that the database in the system of Hayton is a SQL Server, because using existing database management system would eliminate time need for developers to learn/implementing when using database in the system.

As to claim 32, Hayton does not explicitly teach wherein performing the operation, further comprises creating a One-Way Binding to a Dynamically Updated Data Source. However, Hayton teaches certain changes to UI element made by user is not allowed by the server (col. 19, lines 26-34), i.e., Hayton implemented One-Way Binding in the system. It would have been obvious to one of ordinary skill in the art to improve the system of Hayton by creating

One-Way Binding to a Dynamically Updated Data Source because this method would eliminate the step of sending the change to the server application, thus, the performance would be improved.

As to claims 52, 54 and 65, see rejections of claims 17, 21 and 32 above, respectively.

11. Claims 17, 20, 50 and 53 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hayton et al. (U.S. 7,194,743 B2) in view of Marcos et al. (U.S. 6,429,880 B1).

As to claim 17, Hayton does not explicitly teach wherein performing the operation, further comprises representing a single select statement to be submitted to a database. However, Hayton teaches data for use in databinding includes data from database (Application components 34 can be representations of nodes within a structured data file ... structure file; col. 10, lines 20-26). Marcos teaches representing a single select statement to be submitted to a database (When a value is specified for a name ... a select statement is generated; col. 17, lines 42-54). It would have been obvious to one of ordinary skill in the art at the time the invention was made to apply the teaching of Marcos to the system of Hayton because Marco teaches a mechanism for binding Web page definitional elements to a runtime, or back-end, state using a graphical user interface, and bindings can be made between a definitional elements and elements of a database (col. 6, lines 21-33).

As to claim 20, Hayton does not teach wherein performing the operation, further comprises receiving a list of sql commands and names of tables that they should be used to fill. However, Marco teaches receiving a list of sql commands and names of tables that they should be used to fill (Dynamic Web pages can be generated using data retrieved from a database; col. 6, lines 29-33 and col. 16, line 66 – col. 18, line 6). See claim 17 above for reason to apply the teaching of Marcos to the system of Hayton.

As to claims 50 and 53, see rejections of claims 17 and 20 above, respectively.

12. Claims 22 and 55 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hayton et al. (U.S. 7,194,743 B2) in view of Quaeler-Bock et al. (U.S. 6,023,271 B1).

As to claim 22, Hayton does not explicitly teach wherein performing the operation, further comprises allowing resource reference to a transformer class that is defined as code-behind in a current application. However, Hayton teaches transforming values of the data source to values for display on the GUI (With a text type predefined UI element 78c, numeric values may be scaled, stepped and/or clipped; col. 20, lines 35-39). Quaeler-Bock teaches reference a transformer class that is defined as code-behind in a current application (The editor filter interface ... this filtering function can include any combination of transforming, decoding ... as necessary; col. 6, lines 12-18, col. 10, lines 27-43 and the present invention can be implemented in the Java programming language or any other object-oriented language; col. 3, lines 32-35). It would have been obvious to one of ordinary skill in the art at the time the invention was made to

apply the teaching of Quaeler-Bock to the system of Hayton because Quaeler-Bock teaches systems and methods for programming applications with GUIs do not require the time-consuming and error-prone custom coding of GUI/internal variable synchronization routines, in general and GUI/Business Object synchronization routines, in particular (col. 3, lines 12-17).

As to claim 55, see rejection of claim 22 above.

13. Claims 23, 25, 56 and 58 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hayton et al. (U.S. 7,194,743 B2) in view of W3C (XML Path Language (XPath) Version 1.0).

As to claim 23, Hayton does not teach wherein performing the operation, further comprises declaring namespaces to be used in Xml data binding XPath queries. However, Hayton teaches data source can be XML document (Application components 34 can be representations of nodes within a structured data file, such as XML; col. 10, lines 20-26). W3C teaches declaring namespaces to be used in Xml data binding XPath queries (namespace, XPath; see Introduction section; pages 3-5). It would have been obvious to one of ordinary skill in the art at the time the invention was made to apply the teaching of W3C to the system of Hayton because W3C teaches how to apply XML and related information such as namespace and XPath queries to the system of Hayton.

As to claim 25, Hayton does not teach wherein performing the operation, further comprises declaring an individual namespaces within an Extensible Markup Language (XML) data source. However, Hayton teaches data source can be XML document (Application components 34 can be representations of nodes within a structured data file, such as XML; col. 10, lines 20-26). W3C teaches declaring an individual namespaces within an Extensible Markup Language (The namespace declaration; page 5, second and third paragraphs). See claim 23 above for reason to apply the teaching of W3C to the system of Hayton.

As to claims 56 and 58, see rejections of claims 23 and 25 above, respectively.

14. Claims 30 and 63 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hayton et al. (U.S. 7,194,743 B2) in view of Gamma et al. (Design Pattern).

As to claim 30, Hayton does not teach wherein performing the operation, further comprises creating collection view factory classes, which in turn create new CollectionView derived objects. However, Hayton teaches the system can be implemented in Java Programming Language (col. 10, lines 55-65), and the Graphical User Interface elements can be mapped to application components or nodes from XML documents or attributes from database (col. 10, lines 9-26). Gamma teaches Abstract Factory provides an interface for creating families of related or dependent objects without specifying their concrete classes (see Abstract Factory). It would have been obvious to one of ordinary skill in the art at the time the invention was made to apply the teaching of Gamma to the system of Hayton because Gamma teaches how to display

data from difference sources without hard coded, thus it would help in the system of Hayton when the system displays data from different sources: database, XML documents or application objects.

As to claim 63, see rejection of claim 30 above.

Conclusion

15. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. See PTO 892 for related arts.

16. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Diem K. Cao whose telephone number is (571) 272-3760. The examiner can normally be reached on Monday - Friday, 7:30AM - 3:00PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng-Ai An can be reached on (571) 272-3756. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Art Unit: 2194

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

DC

April 12, 2008

/Diem K Cao/

Examiner of Art Unit 2194